arXiv:2210.13723v3 [cs.RO] 28 Dec 2022

S3E: A Large-scale Multimodal Dataset for Collaborative SLAM

Dapeng Feng, Yuhua Qi, Shipeng Zhong, Zhiqiang Chen,
Yudu Jiao, Qiming Chen, Tao Jiang, Hongbo Chen

***Abstract*— With the advanced request to employ a team of robots to perform a task collaboratively, the research community has become increasingly interested in collaborative simultaneous localization and mapping. Unfortunately, existing datasets are limited in the scale and variation of the collaborative trajectories, even though generalization between inter-trajectories among different agents is crucial to the overall viability of collaborative tasks. To help align the research community's contributions with realistic multiagent ordinated SLAM problems, we propose S3E, a large-scale multimodal dataset captured by a fleet of unmanned ground vehicles along four designed collaborative trajectory paradigms. S3E consists of 7 outdoor and 5 indoor sequences that each exceed 200 seconds, consisting of well temporal synchronized and spatial calibrated high-frequency IMU, high-quality stereo camera, and 360 degree LiDAR data. Crucially, our effort exceeds previous attempts regarding dataset size, scene variability, and complexity. It has 4x as much average recording time as the pioneering EuRoC dataset. We also provide careful dataset analysis as well as baselines for collaborative SLAM and single counterparts. Data and more up-to-date details are found at https://github.com/PengYu-Team/S3E.**

I. Introduction

Recently, robotic collaboration has become progressively favored within disparate scenarios, e.g., search and rescue, inventory automation, and agriculture. Collaborative simultaneous localization and mapping (C-SLAM) is at the core of enabling collaboration, increasing the robustness and efficiency of multiagent colocalization and co-mapping of the workspace [1]–[5]. Despite the tremendous progress in C-SLAM during the last decade, reproducibility and benchmarking are still the primary challenges in the C-SLAM research field. Firstly, C-SLAM systems embrace composite software frameworks and many heterogeneous sensors, making them hard to replicate flawlessly. Secondly, the popular approach to evaluating C-SLAM solutions is to split single-agent SLAM datasets into multiple clips and associate each with a virtual agent [6]. However, this method is impractical in real-world multiagent systems since the viewpoints and lighting conditions are the same for all virtual agents at the cutting points. Therefore, systematic evaluation techniques are still lacking for C-SLAM, even though the single-agent SLAM evaluation method [7] is available.

Corresponding author: Yuhua Qi

Dapeng Feng, Yuhua Qi, Shipeng Zhong, Zhiqiang Chen, and Hongbo Chen are with the School of Systems Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China (e-mail: {fengdp5, zhongsp5, chenzq56}@mail2.sysu.edu.cn, {qiyh8, chenhongbo}@mail.sysu.edu.cn).

Yudu Jiao and Tao Jiang are with the School of Automation, Chongqing University, Chongqing 400044, China (e-mail: jiaoyudu@sina.com, jiangtao_1992@cqu.edu.cn).

Qiming Chen is with the College of Materials and Energy, South China Agricultural University, Guangzhou 510642, China (e-mail: chenqiming629@gmail.com).

Fig. 1. **Recording Platforms.** We adopt three remote-control vehicles, i.e., Alpha, Bob, and Carol, to record the collaborative SLAM dataset. Each mobile robot has two high-resolution color cameras, a 360-degree rotating 3D LiDAR, a 9-axis inertial measurement unit, and a dual-antenna real-time kinematic positioning system.

In the SLAM research community, public datasets play an essential role in accelerating the development of SLAM algorithms for real-world applications [8]. On the one hand, the concrete experiment of SLAM involves expensive hardware and perplexed procedures, e.g., calibration and ground truth system. On the other hand, public datasets furnish equitable and impartial benchmarks for comparison and evaluation. Over the last ten years, many SLAM datasets have been available, and each has specific contributions to the availability of multiple sensors and the type of scenarios and motion [9]–[14]. However, most of these datasets focus on the single-agent system equipped with different sensors, e.g., camera, inertial measurement unit (IMU), and LiDAR [10]–[13]. Only a few datasets dedicated to C-SLAM exist [14]–[16] due to the difficulty of data recording increases as the number of the collaborative agents grows up. To further facilitate future research on the C-SLAM, we introduce S3E, a large-scale multimodal C-SLAM dataset, involving images recorded by high-resolution stereo cameras and point clouds scanned by a 3D laser scanner mounted on a fleet of unmanned ground vehicles (UGV), as shown in Figure 1.

Moreover, the inter-trajectories of the collaborative robots system should be designed so that the individual estimation accuracies of the robots can be significantly improved when the evaluated C-SLAM algorithm succeeds. Trajectories of

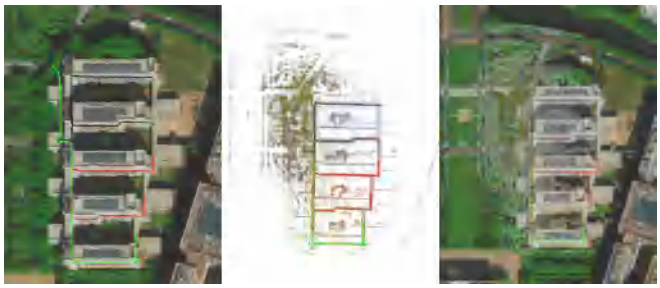


Fig. 2. **The result of the C-SLAM method on the S3E dataset.** The localization trajectories of different agents annotated with different colors lie on the left. The point clouds of the 3D map generated by the C-SLAM method lie in the middle. On the right side, the combination indicates that the C-SLAM method has high-quality localization and mapping abilities.

a multi-robot scenario should be jointly designed according to two principles. The first principle is temporal and spatial diversity. The designed trajectories should consider the different intra/inter-loop closures at different time and places, which provides the complementary views for involved robots. The second one is the communication issue. For practical application, collaborative robots only share information within a certain distance. Hence, the distance between the involved robots in the interaction regions should be in a reasonable range. In this paper, we propose four prototypes of trajectories following different intra/inter-robot closures ethics to evaluate the generalization of C-SLAM methods. Following the proposed trajectory ethics, data was recorded across various conditions in outdoor and indoor environments. Based on our S3E dataset, we conduct numerous experiments with state-of-the-art SLAM methods, including single robot SLAM and collaborative counterparts, using various sensor messages. The results show that existing collaborative SLAM methods perform poor inter-loop closures in at least one sequence. It indicates that boosting the performance and robustness of C-SLAM requires further effort.

In summary, the significant contributions of our works include:

- We collected large-scale C-SLAM datasets with three ground robots. We mounted a top 16-beam 3D laser scanner on each robot, two high-resolution color cameras, a 9-axis IMU, a dual-antenna RTK receiver.
- We recorded the long-term sequences based on the four well-designed trajectory paradigms, considering different intra/inter-robot loop closures situations. To our best knowledge, it is the first multimodal C-SLAM dataset for LiDAR, visual, and inertial data in various indoor and outdoor environments.
- We evaluated state-of-the-art C-SLAM and its single counterparts using different sensor recordings and analyzed the characteristics of different methods.

II. Related Work

State-of-the-art SLAM Using various sensing technologies is necessary when SLAM aims to ensure maximum data integrity in case of system failure. Moreover, the camera and LiDAR are complementary in that each surpasses the limitation of the other. There are two primary coupling strategies to fuse the images and point clouds. The first tactic is loosely-coupled, allowing each modality to run separately and exchange relevant messages in an optimal front-end fusion [17]. The second tactic is tightly-coupled, combining the two modalities at the data representation level in one optimal framework [18], [19]. However, the more various the hardwares are, the more challenges are faced with handling the exact time alignment and complex spacial transformations. Hence, we believe that considering sensor specifications while designing a multimodal SLAM system is a feasible research orientation and will notably improve the robustness and accuracy of the limited computation resource systems.

Collaborative SLAM Many tasks can be performed more rapidly and more efficiently using collaborative agents instead of a single one. As shown in Figure 2, C-SLAM allows a fleet of robots explores different regions of the target simultaneously, which is more efficient than the single agent counterpart. According to network topology, C-SLAM can be divided into centralized, distributed, and hybrid. In centralized conditions, a central server gathers all the incoming data for consistent multi-trajectory estimation and scatters the result to the agents [4], [22]–[24]. While the computation performed by more than one agent in the team is a distributed system. Some distributed systems broadcast generated messages [25]. The above methods are challenging with the increasing number of robots since bandwidth constraints and limited communication range. A flexible solution is to transmit information between agents only within the communication range [2], [5], [26]. Each agent can only access a local map and trajectory from its neighbors to achieve consistent multi-trajectory estimation. In addition, some hybrid solutions are also applicable, where some robots process data and some use the results only [27].

SLAM Datasets. High-quality and large-scale datasets are essential for SLAM research. In recent years, there have been many efforts to release datasets to the community. TartanAir [28] utilizes the high-fidelity visual and physical simulator AirSim [29] to generate a photo-realistic synthetic dataset. Nonetheless, most public datasets were built in the real world, and various environments have been covered. KITTI [12] is the famous sizeable benchmark, offering high-quality urban data and supporting vision-based and LiDAR-related SLAM. However, KITTI does not provide well-synchronized data but only estimates the time delay between different sensors. EuRoC [10] collected the visual and inertial data synchronized with the hardware clock in the machine hall and Vicon room using a micro aerial vehicle to address this issue.

Nevertheless, only a few datasets focus on multiagent scenarios. UTIAS [14] is the pioneering work in the multi-agent SLAM domain, developing a five robots collaborative system. Each robot explored a $15m \times 8m$ area with 15 uniquely identifiable landmarks. However, the sequences in UTIAS acquired in identical environments can be used to evaluate such algorithms by synchronizing the trajectories

TABLE I

COMPARISON OF SOME POPULAR SLAM DATASETS. "SW" MEANS SOFTWARE SYNCHRONIZATION; "HW" MEANS HARDWARE SYNCHRONIZATION.

Dataset	Sensors			Platform	Time Sync.		Avg. Duration[s]	Environment		Ground Truth
	Camera	IMU	LiDAR		Intra-	Inter-		Indoor	Outdoor	
UTIAS [14]	✓	✗	✗	5 UGVs	Sw	NTP [20]	1911.1	✓	✗	Motion Capture
KITTI [12]	✓	✓	✓	Car	Sw	✗	210.9	✗	✓	IMU-RTK
EuRoC [10]	✓	✓	✗	UAV	Hw	✗	125.0	✓	✗	Motion Capture
AirMuseum [16]	✓	✓	✗	3 UGVs, 1UAV	Sw	NTP	351.2	✓	✗	SfM
FordAV [15]	✓	✓	✓	3 Cars	-	GNSS	741.1	✗	✓	IMU-RTK
S3E(ours)	✓	✓	✓	3 UGVs	Hw	GNSS, PTPv2 [21]	459.1	✓	✓	RTK, Motion Capture

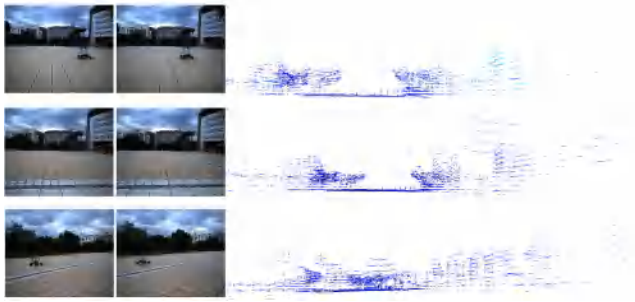


Fig. 3. **The data example of our S3E dataset.** Each line shows the stereo images and point clouds captured by the different mobile robots simultaneously. The qualitative results show that our system performs well intra-synchronization and inter-synchronization.

beforehand. Therefore, the resulting scenarios are unsuitable for evaluating C-SLAM algorithms relevantly. Hence, AirMuseum [16] introduces a heterogeneous multi-robot SLAM dataset in a $40m \times 80m$ warehouse with the ground-truth trajectories computed using Structure-from-Motion (SfM) [30] and constrained with fixed beacons, which is the shortcut solution to the SLAM problems [31]. FordAV [15] presents a multiple autonomous vehicles dataset augmented with a 3D map, collecting seasonal variation in dynamic urban environments but with the same trajectory.

See Table I for a comparison of different datasets.

III. COLLABORATIVE SLAM

The C-SLAM problem can be viewed as a simple extension of the single-agent SLAM when all agents' initial states can be estimated. Moreover, acquiring all the agents' states and observations through the additional inter-measurements linking different agents' maps is easy. For example, given two agents (α, β), the C-SLAM problem can be formulated as:

$$\begin{aligned}\left(X_\alpha^*, X_\beta^*\right) &= \underset{X_\alpha, X_\beta}{\arg\max}\; p\left(X_\alpha, X_\beta | O_\alpha, O_\beta, O_{\alpha\beta}\right), \\ &= \underset{X_\alpha, X_\beta}{\arg\max}\; p\left(O_\alpha, O_\beta, O_{\alpha\beta} | X_\alpha, X_\beta\right) p\left(X_\alpha, X_\beta\right),\end{aligned} \tag{1}$$

where X_α and X_β indicate the initial state of α and β, O_α and O_β are the observations captured by α and β independently, $O_{\alpha\beta}$ is the inter-measurements containing relative pose estimates between α and β, and X_α^*, X_β^* are the estimated poses.

However, when the relative initial states of agents are unknown, there are infinite possible initial alignments between the multiple agent trajectories. Hence, in the absence of a prior distribution $p\left(X_\alpha, X_\beta\right)$, the C-SLAM degrades into the Maximum Likelihood Estimation (MLE).

$$\left(X_\alpha^*, X_\beta^*\right) = \underset{X_\alpha, X_\beta}{\arg\max}\; p\left(O_\alpha, O_\beta, O_{\alpha\beta} | X_\alpha, X_\beta\right). \tag{2}$$

Furthermore, the difference between the global and local viewpoints is the essential discrepancy between C-SLAM and single-robot SLAM [6]. Generally, single-robot SLAM considers the local viewpoint, where the localization and mapping can be expressed in a local inertial reference frame with the origin at the start point of the robot's mission. However, C-SLAM considers the global viewpoint, i.e., each robot's localization and mapping are expressed in a global shared reference frame, e.g., the landmarks can be located within the global coordinates system. Hence, the collaborative robots can collectively perceive the whole environment by establishing the global reference frame through C-SLAM.

IV. THE S3E DATASET

The sample of our S3E dataset is present in Figure 3. It is captured simultaneously with a well time synchronization by different sensors mounted on the three mobile platforms. And we detail the S3E dataset as follows.

A. Hardware Configuration

The data collection was conducted by three categories of sensing modalities operating with different ranges and noise levels. The properties of the sensors and ground truth device are shown in Table II. All the sensors are mounted on *Agilex Scout Mini,* an all-terrain high-speed remote-control mobile platform with four-wheel drive and a maximum speed of $10km/h$. Specifically, we use a *Velodyne VLP-16 Puck* to record the 360-degree long-range point cloud with the strongest return of each laser pulse. Furthermore, we utilize the two mounted *HikRobot MV-CS050-10GC* GigE cameras to capture the stereo-visual data. The baseline of the stereo cameras is $360mm$. The camera images are captured with global shutter scanning and downsampled to 1224×1024 from raw images. Moreover, we use a 9-axis *Xsens MTi-30-2A8G4* IMU to record three accelerometers and three gyrometers. The IMU combines information from the motion model and temporal measurements integration to infer the position (x, y, z) and the orientation $(\theta_x, \theta_y, \theta_z)$ relative to the previous position. For validation and testing, we adopt a real-time kinematic positioning system (RTK) *Femtomes*

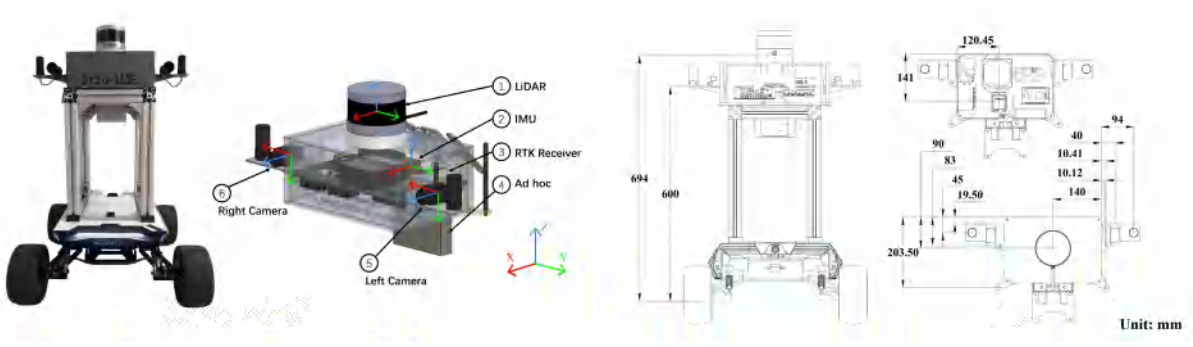


Fig. 4. **Sensor layout and coordinate systems.** The left side is our mobile platform. The coordinates of different sensors lay in the middle. And the right side shows the locations of disparate devices.

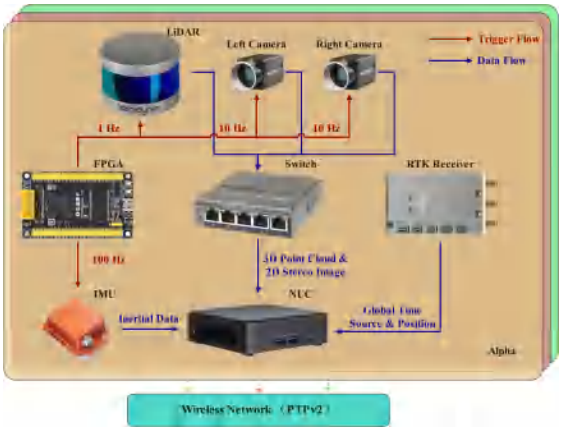


Fig. 5. **FPGA-based synchronization system architecture.** The synchronization system consists of an FPGA as the pulse generator and a NUC computer as the host. For intra-synchronization, FPGA generates four trigger signals with 1 Hz, 10 Hz, and 100 Hz pulses and sends them to the LiDAR, stereo camera, and IMU, respectively. Furthermore, the sensors return the captured data after receiving the trigger signal. For inter-synchronization, we use the GNSS as a global time source in an outdoor situation while utilizing the PTPv2 to align the time source through the wireless network in the GNSS-less environment.

Nano-D equipped with dual-antennas to capture the ground truth in the GNSS available area for validation and testing. The ground truth is collected at 1 Hz. See Figure 4 for the location of all the sensors mounted on the mobile robot platform.

B. Sensors Synchronization

This section describes the time synchronization and sensor calibration, which are essential to use sensor fusion best and achieve the highest system performance in a dynamic multimodal system.

1) Time Synchronization: Our synchronization system uses an *Altera EP4CE10* board as the trigger and an *Intel NUC11TNKv7* as the host, as shown in Figure 5. Critically, our system is equipped with rich I/O interfaces required by different sensors (Table II). For inter-agent synchronization, we divide the issue into two cases for discussion. First, we apply the GNSS time as the global time source in outdoor scenarios to align the agents' timer. Second, in the GNSS-less system (e.g., indoor scenarios), all the agents run the time alignment program to fetch the external global time data from the PTP server through the wireless network. For intra-agent synchronization, the trigger unit periodically generates a pulse to trigger the LiDAR, stereo cameras, and IMU. Significantly, the FPGA generates a 1 Hz pulse to trigger the LiDAR, and then the LiDAR returns 10 Hz data and refreshes the inner counter register after receiving the trigger signal. The cameras and the IMU yield return the data immediately after receiving a trigger pulse. In our case, the trigger unit generates 10 Hz pulses for the cameras and 100 Hz for the IMU. All sensor readings' timestamps are obtained and packed at the host, respectively.

TABLE II

THE PROPERTIES OF THE SENSORS AND GROUND TRUTH DEVICES

Device	Type	Freq.	Characteristics
LiDAR	Velodyne VLP-16 Puck	10 Hz	16 channels, $100m$ range, $360°H \times 30°V$ FOV, $\pm 3cm$ accuray
Stereo Camera	HikRobot MV-CS050-10GC	10 Hz	1224×1024, Color, GigE, lens $8mm, F2.8, 2/3''$
IMU	Xsens MTi-30-2A8G4	100 Hz	9 axis, $450°/s$ gyro, $200m/s^2$ accel, $\pm 8 \times 10^{-4}T$ mag
RTK-GNSS Receiver	Femtomes Nano-D	1 Hz	Horizontal $10mm + 1ppm$, Vertical $15mm + 1ppm$

2) Sensor Calibration: Accurate intrinsic and extrinsic sensors calibration is critical to achieving the highest system performance. The layout of sensors relevant to the dataset are shown in Figure 4. All coordinate system of the sensors follow the right-hand rule. We execute the intrinsic calibration of the cameras using a standard checkerboard calibration. For LiDAR and IMU, the respective manufacturer conducted intrinsic sensor calibration. We apply LiDAR's proprietary calibration model to directly correct the data at the time of capture. After obtaining the intrinsic parameters of cameras and LiDAR, we perform the stereo camera joint calibration and LiDAR-camera joint calibration [32], [33]. Moreover, we carry out the joint calibration between the camera and IMU after modeling the noise of IMU using the Allan standard deviation [34]. All the intrinsic and extrinsic files will be released within the dataset.

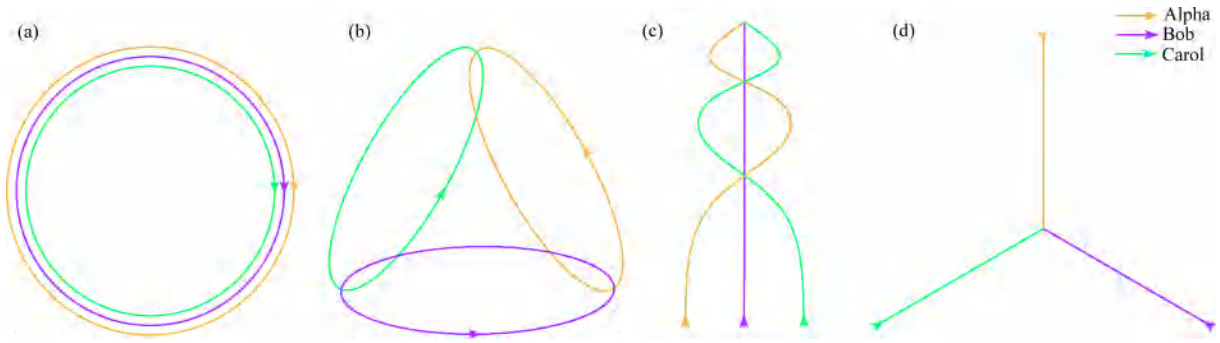


Fig. 6. **Trajectory Paradigms.** Alpha, Bob, and Carol stand for three agents. (a) **Concentric circles.** The agents run in the same ring. (b) **Intersection circles.** The agents run in different rings with the interactions pairwisely. (c) **Intersection curves.** The agents run forward in the different curves with the interactions in the same scenes. (d) **Rays.** The agents start from different sites and only meet in the same place in the end.

TABLE III
ANALYSIS OF S3E DATASET.

Scenario		Time[s]	Trajectory				Ground Truth	Length[m]			Size[GB]
Env.	Region		a	b	c	d		Alpha	Bob	Carol	
Outdoor	Square_1	460			✓		RTK	546.0	496.5	529.2	17.8
	Square_2	255				✓	RTK	-	250.6	246.4	9.4
	Library	454	✓				RTK	507.6	517.2	498.9	16.3
	College	878		✓			RTK	920.5	995.9	1072.3	29.4
	Playground_1	298	✓				RTK	407.7	425.6	445.5	8.7
	Playground_2	222			✓		RTK	265.6	315.7	456.4	6.3
	Dormitory	671			✓	✓	RTK	727.0	719.3	721.9	23.5
Indoor	Teaching_Building	798				✓	RTK (start, end)	617.2	734.4	643.4	27.3
	Laboratory_1	292		✓		✓	Motion Capture	147.7	161.5	141.0	9.6
	Laboratory_2	391		✓			Motion Capture	215.3	199.1	160.3	12.7
	Laboratory_3	410		✓			Motion Capture	219.1	202.2	204.2	13.3
	Laboratory_4	380	✓				Motion Capture	173.7	177.2	180.0	12.7

C. Trajectory Paradigms

In the S3E dataset, we jointly design the trajectories of the agents following four different intra/inter-robot loop closures ethics. As shown in Figure 6, we consider different intra/inter-robot loop closures in the C-SLAM scenarios.

The first kind of trajectory is the typical case in C-SLAM applications. The robot formation runs around the target simultaneously, which is beneficial for reconstructing the dense 3D surface model of the target. Moreover, the intra/inter-robot loop closures make the trajectories more complementary. The second trajectory simulates the regional search and rescue tasks. The agent searches in different regions and shares information with others during the interaction. It raises the requirement that the C-SLAM algorithm has a solid intra-robot loop closures ability and an efficient inter-robot loop closures capability in the small common region.

Furthermore, the third trajectory focuses on scenarios that only have inter-robot loop closures. All the agents start from different places and go forward with some meeting points during the path. In the end, all the robots meet in the same place. In the last scenario, the agents start from different places and only in the same site in the end, which is a formidable sample in C-SLAM, because all the robots only meet at the endpoint, providing little information for the algorithms to review the orbits.

Figure 7 shows the tracks of the S3E dataset in outdoor environments. It contains five representative and functional regions in the campus, i.e., square, library, college, playground, and dormitory.

D. Dataset Analysis

S3E dataset has scenarios selected from both outdoor and indoor environments within the campus. See Table III for the distribution. The average duration of the dataset achieves $459.1s$, which supports the long-term evaluation of C-SLAM. Moreover, the dataset contains different designed trajectories mentioned in Section IV-C, which facilitates tests and comparisons of various C-SLAM methods for different interaction situations, i.e., intra/inter-robot loop closure. Significantly, the dataset contains at least one sequence for each designed trajectory. In addition, Dormitory is the hybrid trajectory of the third and fourth categories, and Laboratory_1 is the hybrid trajectory of the second and the fourth designs.

V. EXPERIMENTS

We provide four single-agent SLAM and three C-SLAM baselines on S3E dataset. More experiment details are shown on our project website.

A. Baselines

ORB-SLAM3 [35] is a tightly visual-inertial system with fast and accurate IMU initialization and multisession map-merging functions.

TABLE IV

BASELINE ATE FOR SINGLE SLAM AND C-SLAM IN THE OUTDOOR ENVIRONMENT. α, β, AND γ STAND FOR ALPHA, BOB, AND CAROL RESPECTIVELY.

Methods	Square_1			Square_2			Library			College			Playground_1			Playground_2			Dormitory		
	α	β	γ	α	β	γ	α	β	γ	α	β	γ	α	β	γ	α	β	γ	α	β	γ
ORB-SLAM3 [35]	**1.16**	15.47	X†	-	2.81	X	11.42	X	X	X	55.47	X	0.87	X	X	3.29	X	X	X	X	X
VINS-Fusion [36]	1.81	X	4.83	-	1.51	0.62	7.95	7.86	5.56	X	16.48	X	3.24	7.36	4.31	31.26	X	X	6.67	3.97	7.70
LIO-SAM [37]	1.19	1.75	X	-	0.73	0.36	1.12	1.52	**1.14**	2.06	3.25	2.43	X	X	0.86	X	X	**0.68**	0.63	1.44	0.91
LVI-SAM [38]	1.21	0.88	X	-	0.79	0.40	1.89	1.67	1.31	2.44	3.14	**1.30**	X	1.59	0.76	6.78	6.10	0.72	0.86	1.48	0.94
COVINS [4]	1.63	**0.83**	1.75	Failed			X			18.27	9.61	54.73	X			X			X		
DiSCo-SLAM [5]	Failed‡			Failed			0.74	1.33	1.27	2.21	**1.35**	Failed	**0.27**	**0.37**	0.38	X			0.54	1.47	Failed
DCL-SLAM [39]	1.23	0.86	**0.66**	-	**0.50**	**0.26**	**0.58**	**1.26**	1.17	**1.51**	1.48	1.87	0.33	0.40	**0.32**	X			**0.52**	**1.37**	**0.51**

† X means that the SLAM method fails to initialize or track frames.
‡ If inter-loop closures detection fails, we mark it "Failed".

Fig. 7. **The outdoor tracks of the S3E dataset.** The tracks of Alpha, Bob, and Carol in outdoor environments are annotated with Orange, Purple, and Cyan.

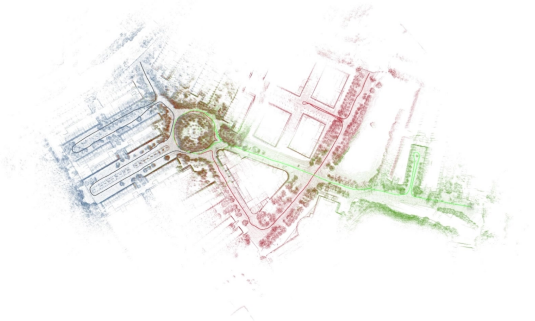


Fig. 8. **The mapping animations of DCL-SLAM [39] using the Dorimatory sequence in the outdoor environment.** In the beginning, each robot performs the self-SLAM according to their independent observations. After then, the collaborative robots establish the global coordinates through the C-SLAM method with the shared pieces of information and transform the local viewpoint to the global one. ***Adobe Reader / KDE Okular*** is recommended for watching the animations.

VINS-Fusion [36] is is an optimization-based state estimator, which supports multiple visual-inertial sensor types.

LIO-SAM [37] formulates a tightly-coupled lidar inertial odometry atop a factor graph, allowing relative and absolute measurements incorporated from different sources as factors into the system.

LVI-SAM [38] proposes a tightly-coupled lidar-visual-inertial odometry, consisting of two sub-systems, i.e., a lidar-inertial odometry and a visual-inertial odometry. It can increase the robustness in both feature-less and texture-less environments when one of the two sub-systems fails.

COVINS [4] develops a centralized visual-inertial C-SLAM system on a powerful server, which gathers the contributed data generated by ORB-SLAM3 [35] from multiple agents and estimates an accurate global optimization. It refines the joint estimate by employing place recognition and removing redundant data.

DiSCo-SLAM [5] is the pioneering distributed multi-robot C-SLAM framework for 3D LiDAR observations using a lightweight scan context descriptor [40]. It requires a comparatively low communication bandwidth for broadcasting the exchanged information.

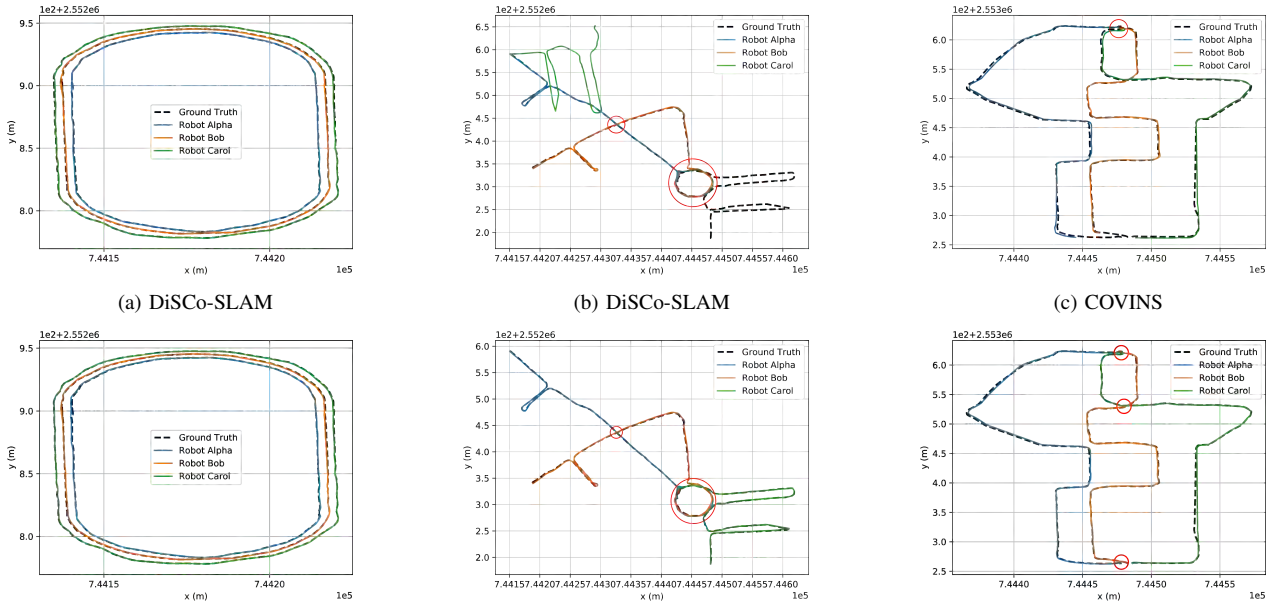
DCL-SLAM [39] consists a keyframe-based distributed loop closure detector and a distributed pose graph optimizer, supporting various front-end feature extractors. In our case, we adopt LIO-SAM [37] as the front-end extractor.

For most of the baselines, we only modify the intrinsic and extrinsic of the sensors and use the left camera for evaluation.

B. Metrics

Absolute Translation Error (ATE) is widely used to evaluate the SLAM's performance, which only requires a single predicted number to calculate. Given the ground truth Q_i and the predicted position P_i, we can estimate the similarity transformation matrix $S \in Sim(3)$ with least squares estimate (LSE) and calculate the ATE for the n points trajectories using Equations (3).

$$F_i = Q_i^{-1} S P_i,$$
$$ATE = \left(\frac{1}{n} \sum_{i=1}^{n} ||trans(F_i)||^2 \right)^{\frac{1}{2}}. \tag{3}$$



(a) DiSCo-SLAM (b) DiSCo-SLAM (c) COVINS

(d) DCL-SLAM (e) DCL-SLAM (f) DCL-SLAM

Fig. 9. **The qualitative results of C-SLAM.** The dashed curves present the ground truth, and the predicted trajectories are drawn with the colored solid curve. The red circles indicate that the C-SLAM successfully detected the inter-loop closures.

C. Ground Truth

We adopt three methods to generate the ground truth tracks. In the GNSS available regions, we provide the centimeter-level localization ground truth recorded by the dual-antenna RTK device. For the GNSS-less scenarios, e.g., the robots move inside the building, we record the start and end points of the orbits outside the building with the RTK device. Furthermore, we utilize the motion capture device to record the agents' start and end points in the indoor case. More details about the ground truth and evaluation are present on our project website.

D. Results

Table IV summarizes our results in the outdoor environment. For single-robot SLAM, the LiDAR-based approaches generally outperform vision-based methods. LVI-SAM [38] increase the robustness because it develops a multimodal system. Most of the vision-based methods failed to track frames when cornering. Hence, the LiDAR-based C-SLAM exceeds the visual-based C-SLAM since its front-end extractor is more robust than the visual-based one. Significantly, we observed that the C-SLAM increased the robustness and accuracy of state estimation when it succeeded in detecting the inter-loop closures. For example, DCL-SLAM [39] achieves an average ATE reduction of 0.42 compared to its front end, LIO-SAM [37]. In Playground_1, the concentric circles, both Alpha and Bob fail to track frames when deploying the single-agent SLAM, LIO-SAM. However, they succeeded when collaborating with Carol through DCL-SLAM, the distributed LiDAR-based C-SLAM method. Not only that, in the DCL-SLAM setting, the performance of Carol achieves 0.32, superior to LIO-SAM. A similar phenomenon happened in Square_1, the intersection curves. Compared with LIO-SAM, Carol succeeded in tracking frames with collaboration through DCL-SLAM. Furthermore, in Square_1, COVINS [4] surpasses its single-agent counterpart with an average ATE reduction of 7.09, and Carol achieves 1.75 ATE with the collaboration while it fails in ORB-SLAM3 [35].

As shown in Figure 9, we annotated the estimated trajectories with the red circle when the C-SLAM successfully detected the inter-loop closures. On the left side, DiSCo-SLAM and DCL-SLAM succeed in the easy case since the considerable overlap among different agents' observations guarantees redundant features for back-end optimization. In the middle, DiSCo-SLAM failed to detect the inter-loop closures between Carol and others, while DCL-SLAM successfully detected the inter-loop closures between Carol and other agents at the circle flowerbed with the middle-level overlap between trajectories. The result of a little overlap between the inter-trajectories is shown on the right side. Compared with DCL-SLAM, COVINS successfully detected the inter-loop closures between different agents at the endpoint where two agents move co-directionally but failed to match the features between Bob and Carol at the middle point where two agents move relatively. It is one of the reasons why DCL-SLAM is superior to COVINS. The animations of collaborative localization and mapping base on DCL-SLAM in the outdoor and indoor environments are shown in Figure 8 and Figure 10 respectively. Please find more detail and higher resolution version animations in our supplementary video about our S3E dataset.

The results show that the state-of-the-art C-SLAM approaches may perform well on the inter-trajectories with considerable overlap. The shared region size is the key challenge at the heart of C-SLAM systems, which decides how many matched features are for back-end optimization.

They still require significant improvement to satisfy the low overlap requirement between inter-trajectories in daily life.

VI. Conclusion

We presented a large-scale C-SLAM dataset recorded by three remote-control ground robots. It contains LiDAR-Visual-Interial data. The dataset includes a series of real-world scenarios collected in indoor and outdoor environments with modern sensing technologies and high-quality hardware time-synchronization. We also conduct well calibration between the left camera and other sensors. Our dataset contains various trajectories considering different intra/inter-robot loop closures conditions, which is intended to fill a void of benchmark data in multimodal C-SLAM evaluation. Furthermore, we evaluated the performance of state-of-the-art C-SLAM and its single-robot front-end counterparts on the dataset. We analyzed the effects and limitations of existing C-SLAM approaches in different scenarios according to the proposed trajectory paradigms, which may identify potential developing trends for C-SLAM. In the future, we are intended to extend our formation with the ground and aerial robots to establish a large-scale heterogeneous dataset and maintain an online leaderboard to promote the research on C-SLAM related tasks.

References

[1] M. Karrer and M. Chli, "Distributed variable-baseline stereo slam from two uavs," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, May 2021, pp. 82–88. 1

[2] P.-Y. Lajoie, B. Ramtoula, Y. Chang, L. Carlone, and G. Beltrame, "Door-slam: Distributed, online, and outlier resilient slam for robotic teams," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1656–1663, April 2020. 1, 2

[3] J. Liu, R. Liu, K. Chen, J. Zhang, and D. Guo, "Collaborative visual inertial slam for multiple smart phones," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, May 2021, pp. 11 553–11 559. 1

[4] P. Schmuck, T. Ziegler, M. Karrer, J. Perraudin, and M. Chli, "Covins: Visual-inertial slam for centralized collaboration," in *2021 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, Oct 2021, pp. 171–176. 1, 2, 6, 7

[5] Y. Huang, T. Shan, F. Chen, and B. Englot, "Disco-slam: Distributed scan context-enabled multi-robot lidar slam with two-stage global-local graph optimization," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1150–1157, April 2022. 1, 2, 6

[6] P. Lajoie, B. Ramtoula, F. Wu, and G. Beltrame, "Towards collaborative simultaneous localization and mapping: a survey of the current research landscape," *CoRR*, vol. abs/2108.08325, 2021. 1, 3

[7] M. Bujanca, P. Gafton, S. Saeedi, A. Nisbet, B. Bodin, M. F. O'Boyle, A. J. Davison, P. H. Kelly, G. Riley, B. Lennox, M. Luján, and S. Furber, "Slambench 3.0: Systematic automated reproducible evaluation of slam systems for robot vision challenges and scene understanding," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 6351–6358. 1

[8] L. Meyer, M. Smíšek, A. Fontan Villacampa, L. Oliva Maza, D. Medina, M. J. Schuster, F. Steidle, M. Vayugundla, M. G. Müller, B. Rebele, A. Wedler, and R. Triebel, "The madmax data set for visual-inertial rover navigation on mars," *Journal of Field Robotics*, vol. 38, no. 6, pp. 833–853, 2021. 1

[9] M. Ferrera, V. Creuze, J. Moras, and P. Trouvé-Peloux, "Aqualoc: An underwater dataset for visual–inertial–pressure localization," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1549–1559, 2019. 1

[10] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016. 1, 2, 3

[11] J. Delmerico, T. Cieslewski, H. Rebecq, M. Faessler, and D. Scaramuzza, "Are we ready for autonomous drone racing? the uzh-fpv drone racing dataset," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 6713–6719. 1

[12] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 3354–3361. 1, 2, 3

[13] T. Pire, M. Mujica, J. Civera, and E. Kofman, "The rosario dataset: Multisensor data for localization and mapping in agricultural environments," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 633–641, 2019. 1

[14] K. Y. Leung, Y. Halpern, T. D. Barfoot, and H. H. Liu, "The utias multi-robot cooperative localization and mapping dataset," *The International Journal of Robotics Research*, vol. 30, no. 8, pp. 969–974, 2011. 1, 2, 3

[15] S. Agarwal, A. Vora, G. Pandey, W. Williams, H. Kourous, and J. McBride, "Ford multi-av seasonal dataset," *The International Journal of Robotics Research*, vol. 39, no. 12, pp. 1367–1376, 2020. 1, 3

[16] R. Dubois, A. Eudes, and V. Frémont, "Airmuseum: a heterogeneous multi-robot dataset for stereo-visual and inertial simultaneous localization and mapping," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Sep. 2020, pp. 166–172. 1, 3

[17] L. Andresen, A. Brandemuehl, A. Honger, B. Kuan, N. Vödisch, H. Blum, V. Reijgwart, L. Bernreiter, L. Schaupp, J. J. Chung, M. Burki, M. R. Oswald, R. Siegwart, and A. Gawel, "Accurate mapping and planning for autonomous racing," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2020, pp. 4743–4749. 2

[18] J. Lin and F. Zhang, "R3live: A robust, real-time, rgb-colored, lidar-inertial-visual tightly-coupled state estimation and mapping package," in *2022 International Conference on Robotics and Automation (ICRA)*, May 2022, pp. 10 672–10 678. 2

[19] W. Chen, Y. Wang, H. Chen, and Y. Liu, "Eil-slam: Depth-enhanced edge-based infrared-lidar slam," *Journal of Field Robotics*, vol. 39, no. 2, pp. 117–130, 2022. 2

[20] D. Mills, "Internet time synchronization: the network time protocol," *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, Oct 1991. 3

[21] V. Shankarkumar, L. Montini, T. Frost, and G. Dowd, "Precision time protocol version 2 (ptpv2) management information base," *RFC*, vol. 8173, pp. 1–64, 2017. 3

[22] R. Dubé, A. Cramariuc, D. Dugas, H. Sommer, M. Dymczyk, J. Nieto, R. Siegwart, and C. Cadena, "Segmap: Segment-based mapping and localization using data-driven descriptors," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 339–355, 2020. 2

[23] K. Ebadi, Y. Chang, M. Palieri, A. Stephens, A. Hatteland, E. Heiden, A. Thakur, N. Funabiki, B. Morrell, S. Wood, L. Carlone, and A.-a. Agha-mohammadi, "Lamp: Large-scale autonomous mapping and positioning for exploration of perceptually-degraded subterranean environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, May 2020, pp. 80–86. 2

[24] P. Schmuck and M. Chli, "Ccm-slam: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams," *Journal of Field Robotics*, vol. 36, no. 4, pp. 763–781, 2019. 2

[25] T. Cieslewski, S. Choudhary, and D. Scaramuzza, "Data-efficient decentralized visual slam," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 2466–2473. 2

[26] S. Choudhary, L. Carlone, C. Nieto, J. Rogers, H. I. Christensen, and F. Dellaert, "Distributed mapping with privacy and communication constraints: Lightweight algorithms and object-based models," *The International Journal of Robotics Research*, vol. 36, no. 12, pp. 1286–1311, 2017. 2

[27] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza, "Collaborative monocular slam with multiple micro aerial vehicles," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 3962–3970. 2

[28] W. Wang, D. Zhu, X. Wang, Y. Hu, Y. Qiu, C. Wang, Y. Hu, A. Kapoor, and S. Scherer, "Tartanair: A dataset to push the limits of visual slam," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2020, pp. 4909–4916. 2

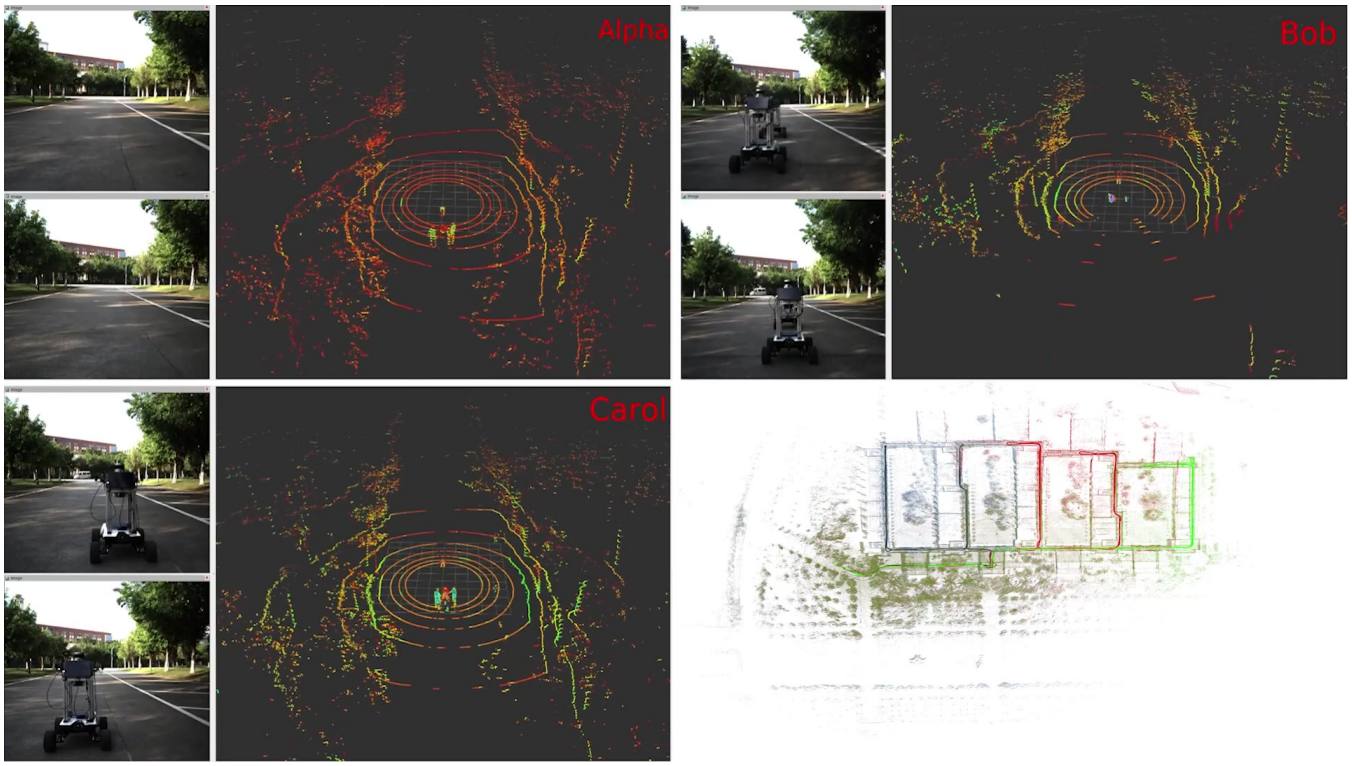


Fig. 10. **The mapping animations of DCL-SLAM [39] using the Teaching Building sequence in the indoor environment.** The layout consists of four parts. The animations of collaborative localization and mapping procedure lie in the bottom right corner. The other three parts are the local observations, e.g., stereo images and 3D laser points, of different robots, i.e., Alpha, Bob, and Carol. ***Adobe Reader / KDE Okular*** is recommended for watching the animations.

[29] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*, M. Hutter and R. Siegwart, Eds. Cham: Springer International Publishing, 2018, pp. 621–635. 2

[30] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 4104–4113. 3

[31] B. Pfrommer and K. Daniilidis, "Tagslam: Robust SLAM with fiducial markers," *CoRR*, vol. abs/1910.00679, 2019. 3

[32] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov 2000. 4

[33] L. Zhou, Z. Li, and M. Kaess, "Automatic extrinsic calibration of a camera and a 3d lidar using line and plane correspondences," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 5562–5569. 4

[34] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 1280–1286. 4

[35] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, Dec 2021. 5, 6, 7

[36] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," *CoRR*, vol. abs/1901.03638, 2019. 6

[37] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2020, pp. 5135–5142. 6, 7

[38] T. Shan, B. Englot, C. Ratti, and D. Rus, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, May 2021, pp. 5692–5698. 6, 7

[39] S. Zhong, Y. Qi, Z. Chen, J. Wu, H. Chen, and M. Liu, "Dcl-slam: A distributed collaborative lidar slam framework for a robotic swarm," *CoRR*, vol. abs/2210.11978, 2022. 6, 7, 9

[40] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3d point cloud map," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 4802–4809. 6